

Koło Naukowe Robotyków
"KoNaR"
Stanowisko do badań własności sonaru
ultradźwiękowego
Raport z projektu

Karol Sydor
Łukasz Tułacz

1 lipca 2008

Spis treści

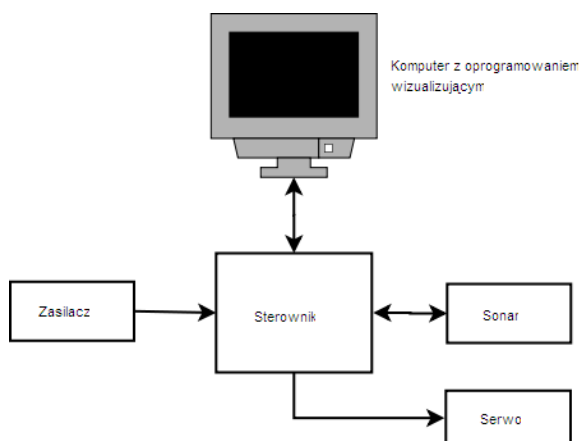
1	Wstęp	2
2	Opis stanowiska	2
2.1	Komputer z oprogramowaniem do wizualizacji	2
2.1.1	Sposób obliczania odległości	2
2.1.2	Funkcje oprogramowania	3
2.2	Sonar	4
2.2.1	Parametry sonaru	5
2.2.2	Komunikacja i obsługa	5
2.2.3	Szczegóły techniczne	6
2.3	Sterownik	8
2.3.1	Opis	8
2.3.2	Komunikacja	8
2.4	Serwo	12
2.5	Zasilacz	13
3	Przekazanie projektu i własność intelektualna	14
4	Materiały dodatkowe	14

1 Wstęp

Celem projektu było praktyczne zrealizowanie stanowiska, umożliwiającego badanie własności sonaru ultradźwiękowego. Zadanie udało się zrealizować. Niniejszy dokument zawiera szczegóły konstrukcyjne i podsumowuje projekt.

2 Opis stanowiska

Stanowisko składa się z 5 modułów (rys.??)



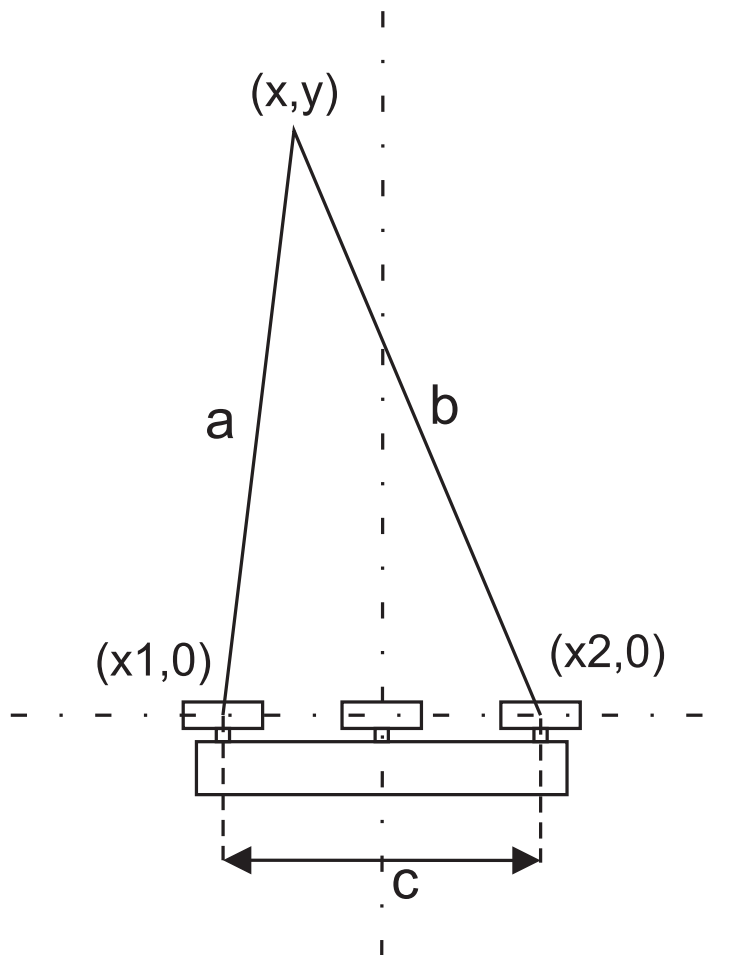
Rysunek 1: Diagram stanowiska

2.1 Komputer z oprogramowaniem do wizualizacji

Komputer klasy PC pełni funkcje odpowiedzialne za wizualizację pomiarów, oraz sterowanie całym stanowiskiem. Oprogramowanie zostało oparte o biblioteki Qt4. Komunikacja ze sterownikiem sonaru wykorzystuje port RS232. Do obsługi portu wykorzystano biblioteki Qextserialport które działają zarówno w środowisku Windows jak i Linux. Stworzono osobny widget przedstawiający efekty rysowania. Do zwizualizowania wyników wykorzystano funkcje biblioteki QPainter natomiast do wyeksportowania wynikowego obrazu do formatu PNG posłużono się klasą QImage a do zapisu obrazu na dysku wykorzystano bibliotekę QFileDialog. W przypadku pomiaru seryjnego kolejny pomiar jest wyzwalany za pomocą funkcji z biblioteki QTimer.

2.1.1 Sposób obliczania odległości

Sterownik przesyła do komputera dwie liczby całkowite odpowiadające odległością zmierzonym przez oba kanały sonaru. Na Rys.?? są one oznaczone jako a i b . Odległości te można przedstawić w postaci dwóch wektorów w układzie współrzędnych o środku w połowie szerokości sonaru ($c/2$). Wektory te rozpoczynają się w punktach $(x_1, 0)$ i $(x_2, 0)$, a kończą w jednym punkcie (x, y) który stanowi reprezentację odległości. Punkty x_1 i x_2 są ustalone na stałe i wynoszą: $x_1 = -c/2$ i $x_2 = c/2$. Do wyznaczenia współrzędnych wykorzystano wzór na



Rysunek 2: Triangulacja

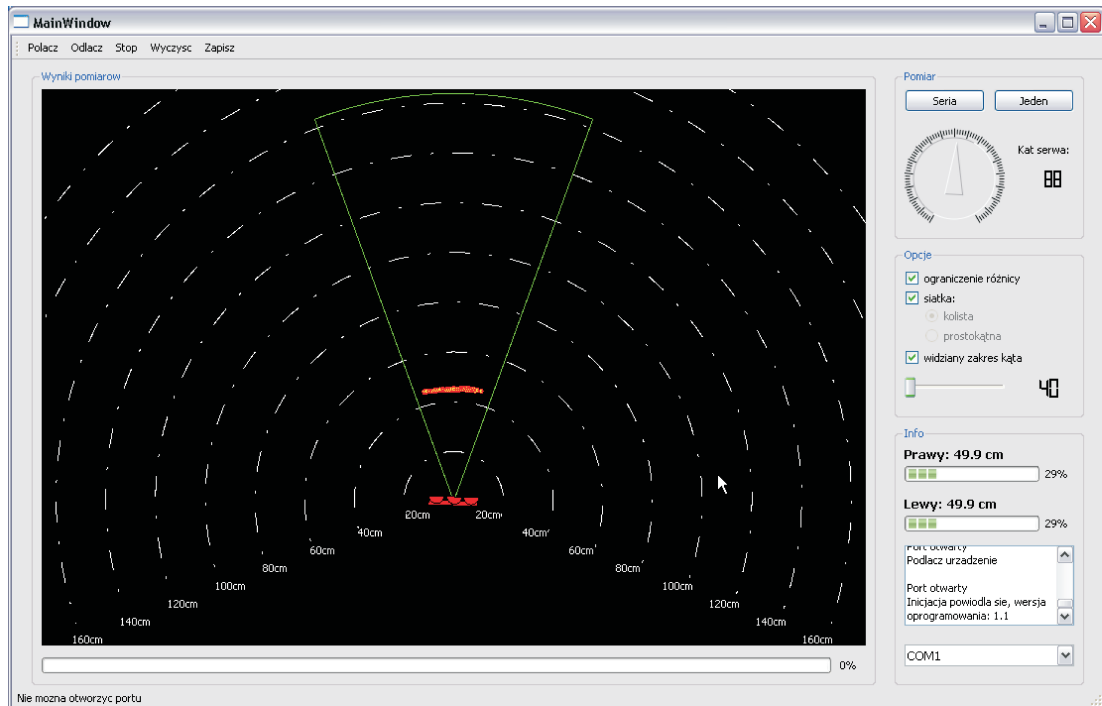
długość wektora. Po przekształceniu i podstawieniu do układu równań otrzymano następujące wzory:

$$\begin{cases} x = \left(\frac{c^2 - a^2 - x_2^2 + x_1^2}{2(x_1 - x_2)} \right) \\ y = \sqrt{a^2 - x_1^2 - \left(\frac{c^2 - a^2 - x_2^2 + x_1^2}{2(x_1 - x_2)} \right)^2 + 2x_1 \left(\frac{c^2 - a^2 - x_2^2 + x_1^2}{2(x_1 - x_2)} \right)} \end{cases}$$

Obliczenia są powtarzane dla każdego pomiaru, a wyliczony punkt jest obracany względem początku układu współrzędnych o aktualny kąt obrotu serwomechanizmu przy danym pomiarze.

2.1.2 Funkcje oprogramowania

- Pomiar seryjny - powoduje rozpoczęcie serii pomiarów i rysowanie mapy terenu. Nie ma reakcji gdy nie otwarto wcześniej żadnego portu.
- Wybór portu - umożliwia wybór portu do którego podłączone jest urządzenie spośród wszystkich dostępnych. Porty są skanowane zaraz po uruchomieniu programu i dodawane do listy wyboru. Po wyborze port jest automatycznie otwierany.
- Odłącz - powoduje zamknięcie portu.
- Stop - powoduje przerwanie pomiaru seryjnego.
- Wyczyść - czyści ekran z punktów reprezentujących odległość.



Rysunek 3: Oprogramowanie wizualizacyjne

- Zapisz - zapisuje obraz do pliku w formacie PNG.
- Pomiar pojedynczy - powoduje narysowanie jednej kropki odpowiadającej pojedynczemu pomiarowi. Nie ma reakcji gdy nie otwarto wcześniej żadnego portu.
- Ręczna nastawa kąta serwomechanizmu - funkcja pomocna przy pojedynczych pomiarach, jest nieaktywna w czasie pomiaru seryjnego.
- Ograniczenie różnicy - jest to funkcja która powoduje ignorowanie odczytu odległości w przypadku gdy różnica między odczytem z lewego i prawego kanału jest większa niż szerokość sonaru(c). Pozwala to uniknąć rysowania błędnych pomiarów.
- Siatka - powoduje nałożenie siatki ze skalą dla łatwiejszego ustalenia odległości przeszkody. Siatkę można nakładać w trakcie trwania pomiaru seryjnego.
- Zakres kąta - umożliwia ograniczenie kąta obrotu serwomechanizmu od 40° do 180° . Ograniczenie to jest widziane na ekranie w postaci wycinku koła. Można wyłączyć efekt wizualny ale to nie powoduje wyłączenia ograniczenia kąta.

2.2 Sonar

Stereofoniczny moduł sonaru (((STEREO))) został zaprojektowany z myślą o zastosowaniu w robotach minisumo. Jego parametry pozwalają wykorzystać go w dowolnej aplikacji, gdzie wymagany jest pomiar odległości w zakresie 5cm-1,8m. Ze względu na zastosowanie dwóch torów odbiorczych, możliwe jest rozróżnienie kierunku z jakiego powraca echo.



Rysunek 4: Moduł sonaru

2.2.1 Parametry sonaru

Parametry sonaru dla wersji oprogramowania 1.1 (parametry wyznaczone dla przetworników $\phi = 12mm$.)

- Interfejs SPI
- Zasilanie 5V, max. 50mA
- Zasięg 5cm-1.8m, poniżej 5cm sonar nie zwraca prawidłowej odległości, ale widzi przeszkodę. (Zasięg ograniczany jest programowo)
- Czas pomiaru do 12,5ms, częstotliwość maksymalna 80pomiarów/s
- Wymiary 64x25mm

2.2.2 Komunikacja i obsługa

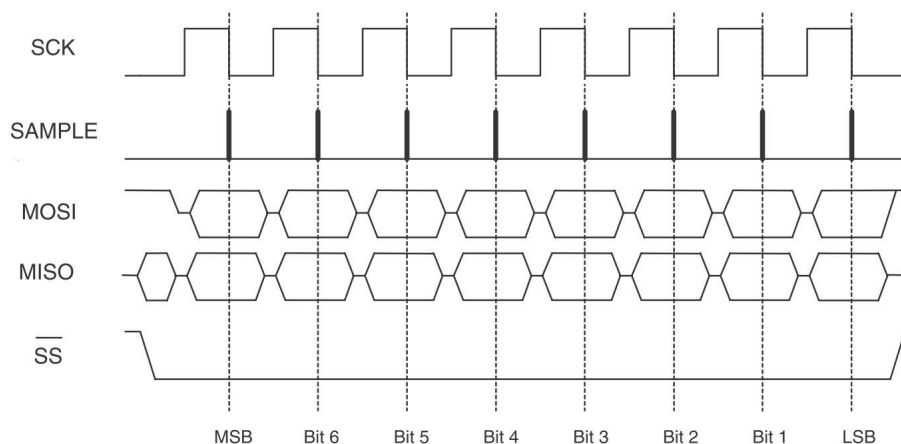
Opis wyprowadzeń sonaru:

Pin	Sygnal
1	Reset
2	SCK
3	MISO
4	MOSI
5	SS
6	GND
7	VCC

Tablica 1: Opis wyprowadzeń

Sonar komunikuje się z jednostką nadrzędną za pomocą interfejsu SPI. Maksymalna częstotliwość zegara SPI to 2MHz. Sonar posiada wejście RESET, jego podłączenie nie jest konieczne. Wyprowadzenie oznaczone jako SS to pin SlaveSelect, występujący częściej pod nazwą ChipSelect. Jest aktywowany stanem niskim. Interfejs należy ustawić w trybie MODE 1: CPHA = 1, CPOL = 0. Bit MSB powinien być transmitowany jako pierwszy. Format transmisji pojedynczego bajtu obrazuje rysunek ???. Każda ramka wysyłana do sonaru powinna się składać z 4 bajtów. W przypadku kiedy sonar nie odbierze echa zwraca

wartość 20000. Od momentu wysłania komendy pomiaru, do jego zakończenia należy odczekać co najmniej 12,5ms.



Rysunek 5: Transmisja pojedynczego bajtu

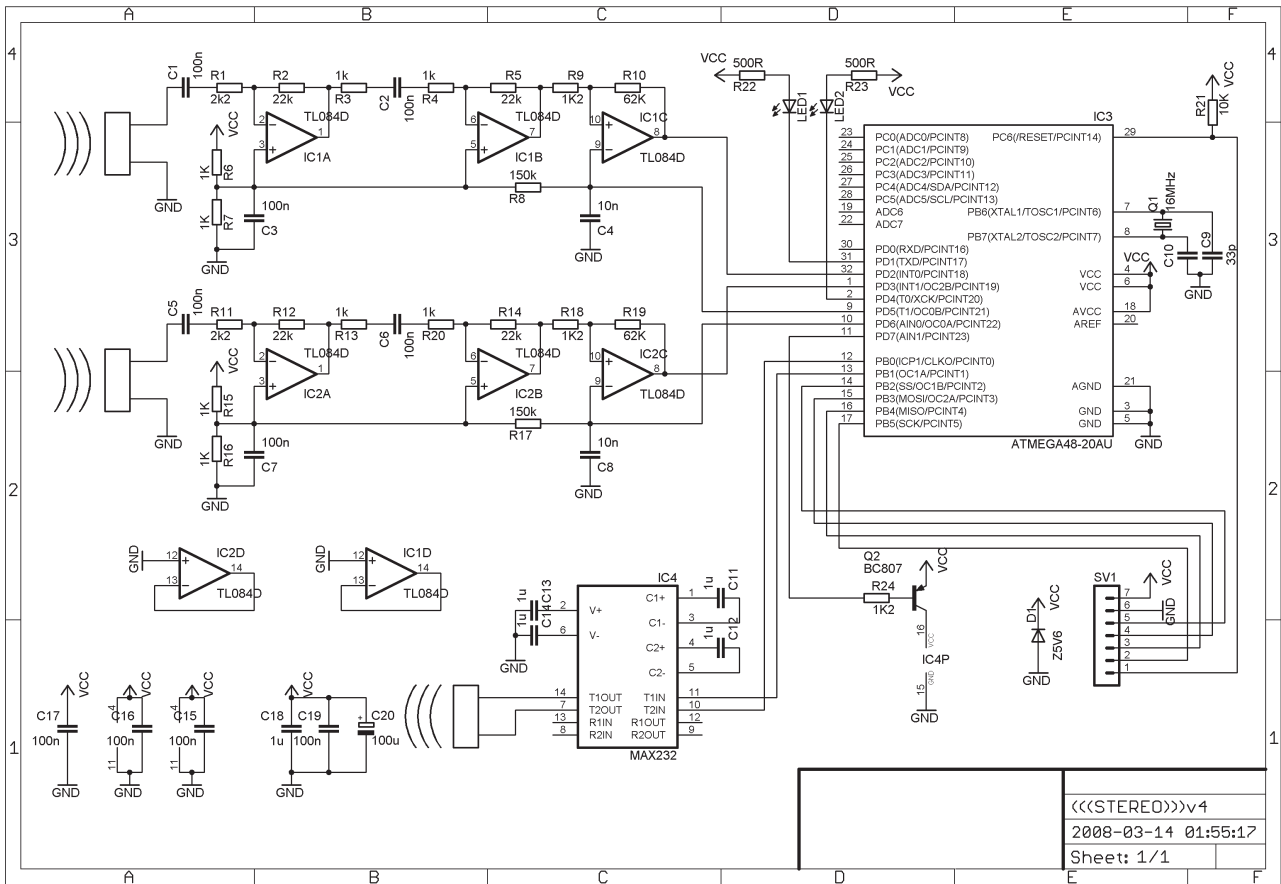
Po podłączeniu sonaru do zasilania należy odczekać co najmniej 100ms. Po tym czasie zaświeci się jedna dioda LED. W tym trybie sonar czeka na inicjację. Polega ona na wysłaniu do sonaru czterech bajtów o wartości 0xF0. Sonar odpowiada 3 bajtami bez znaczenia, oraz czwartym o wartości oznaczającej wersję firmware (10 oznacza wersję 1.0 itp.) Po poprawnej inicjacji zaświecą się obie diody LED. Sonar na każdą ramkę odpowiada 4 bajtami będącymi wynikiem ostatniego pomiaru, w formacie Hi(Lewy) Low(Lewy) Hi(Prawy) Low(Prawy). Gdzie Lewy i Prawy to 16 bitowe wartości pomiaru. Hi(Lewy) to górne 8 bitów, Low(Lewy) to dolne 8 bitów tworzące 16 bitową zmienną. Odległość można wyliczyć ze wzoru ??

$$D = W * 0,09mm. \quad (1)$$

Gdzie D to odległość w mm. W to zmierzona wartość. W przypadku kiedy sonar nie widzi przeszkody w swoim zasięgu zwraca wartość 20000. Po wydaniu komendy pomiaru należy odczekać maksymalnie 12,5ms na wyniki pomiaru.

2.2.3 Szczegóły techniczne

Mikrokontrolerem sterującym modułu sonaru jest procesor ATMEGA48 będący przedstawicielem rodziny AVR. Taktowany jest rezonatorem kwarcowym o częstotliwości 16MHz. Mikrokontroler uruchamia za pomocą tranzystora Q2 układ MAX232 który jest wykorzystany jako wzmacniacz toru nadawczego. Mikrokontroler w celu wykonania pomiaru generuje 8 "strzałek" i od razu rozpoczyna mierzenie czasu. W momencie kiedy procesor wykryje stan wysoki na wyjściach toru odbiorczego, rejestruje czas. Za pomocą wyjść PD5 i PD6 mikrokontroler uruchamia dynamiczne progowanie. Kondensator C4 i C8 sprawia że na wejściu odwracającym komparatora napięcie powoli zbliża się do wartości 2,5V.



Rysunek 6: Schemat sonaru

Tym samym maleje próg komparacji. Odbierany sygnał jest tym słabszy im pochodzi z większej odległości. Progowanie dynamiczne sprawia że sonar jest tym bardziej czuły im większa jest potencjalna odległość od przeszkody. Sygnały z przetworników, dochodzące do komparatora są wzmacniane przez kaskadę dwóch wzmacniaczy. Pierwszy o wzmacnieniu ok.10, drugi ok.20. W celu zapewnienia symetryczności toru wejściowego wprowadzono za pomocą dzielnika R6/R7 i R15/R16 potencjał równy połowie napięcia zasilania który jest napięciem odniesienia dla wzmacniaczy wejściowych (to do jego potencjału zbliża się próg przy komparatorze). Dioda Z1 stanowi zabezpieczenie przed zbyt wysokim napięciem. Dwie niebieskie diody LED gasną w chwili złapania echa i są zapalane przy rozpoczęciu pomiaru. Płytką drukowaną została zaprojektowana w ten sposób, żeby oba torzy posiadały jak najbardziej zbliżone rozmieszczenie ścieżek. Dzięki dynamicznemu progowaniu udało się uzyskać bardzo zbliżone parametry obu kanałów. Ponadto cała część "cyfrowa" i nadawcza została umieszczona po drugiej stronie płytki. Poprzednia wersja tego sonaru (bez progowania dynamicznego) została użyta w robotach minisumo "Eurobot".

2.3 Sterownik

2.3.1 Opis

Sterownik realizuje 3 podstawowe funkcje:

- Komunikacja z komputerem w celu wymiany danych pomiarowych i komend sterujących
- Komunikacja z sonarem
- Generowanie impulsów sterujących dla serwomechanizmu

Sterownik został zrealizowany w oparciu o moduł z mikrokontrolerem Freescale MC9S12A64. Moduł został umieszczony na płycie głównej, na której znajdują się złącza: dla serwa, sonaru, zasilające oraz diód LED (JP1 i JP2). Ponadto znajduje się tam też konwerter poziomów MAX232, wraz ze złączem DIL do podłączenia gniazda DB9 (zaciśniętego na taśmę), oraz kondensatory zapewniające odpowiednią filtrację napięć zasilających. Płytką została zaprojektowana w ten sposób że masa zasilania sonaru jest połączona tylko w jednym punkcie z płaszczyzną masy sterownika. Takie rozwiązanie stosuje się w celu eliminacji zakłóceń.

2.3.2 Komunikacja

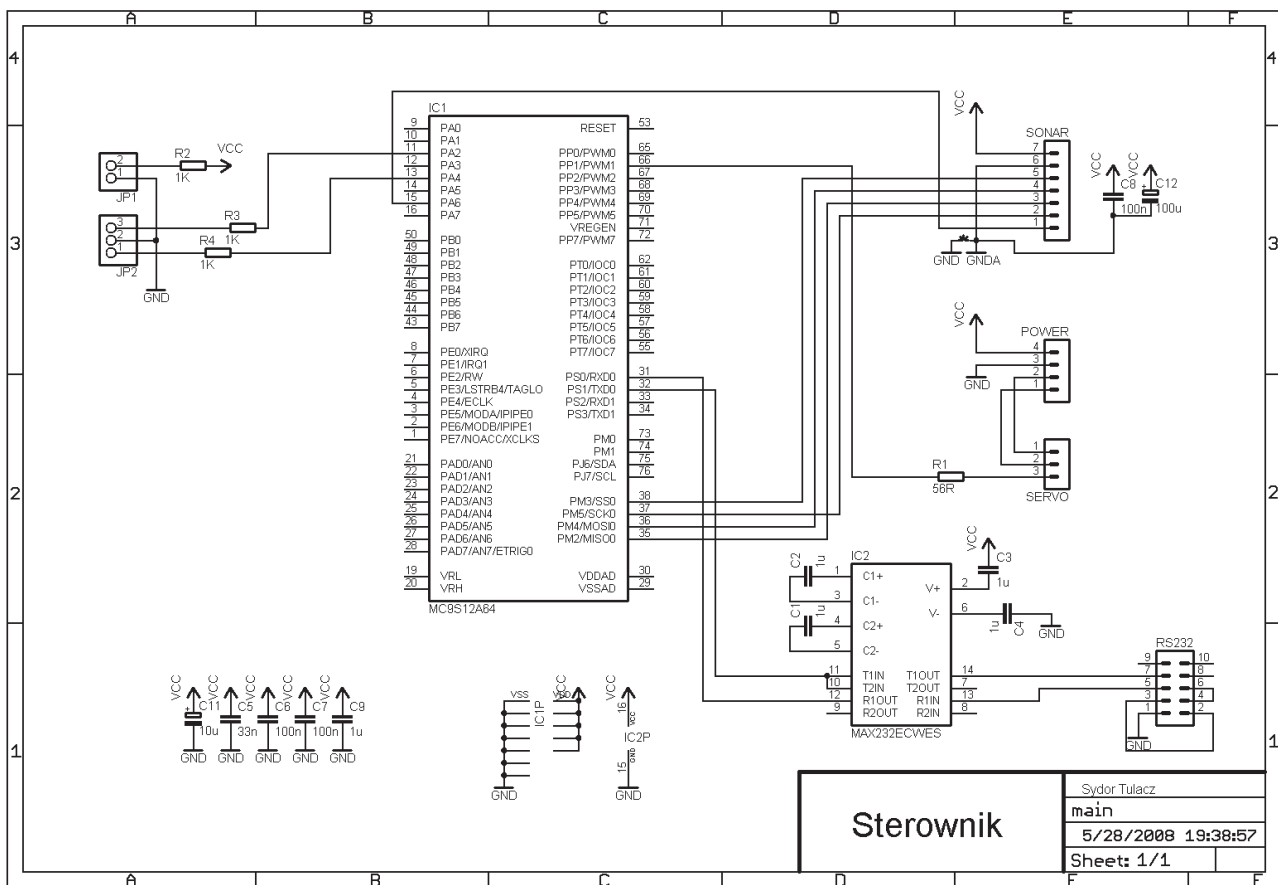
Komunikacja odbywa się za pomocą interfejsu RS232C. w trybie asynchronicznym, półduplexowym bez potwierdzeń. Na płycie sterownika dokonano połączeń sygnałów CTS z RTS oraz DSR z DTR (sygnały będące standardem interfejsu RS232C). Powoduje to że komputer zawsze jest gotowy do odbierania i wysyłania danych, więc można "na sztywno" zaprogramować sposób komunikowania się komputera z urządzeniem pomiarowym. Schemat połączenia został przedstawiony na rys.??

Przyjęto że wymiana danych między komputerem a sterownikiem odbywa się według następującej konwencji: Komputer jest jednostką nadrzędną i tylko on może rozpocząć transmisję. Wysyła ramkę do sterownika po czym czeka odpowiednią ilość czasu potrzebną na przetworzenie informacji przez sterownik. Następnie przechodzi w stan odbierania danych. Sterownik jest tak zaprogramowany aby po każdym odebraniu ramki odsyłać przetworzoną ramkę. Po zakończeniu wysyłania znów przechodzi w stan odbierania i czeka na ramkę od komputera. Prędkość transmisji wynosi 56700baud, z jednym bitem startu i stopu, bez bitu parzystości.

Ramka Ramka składa się z czterech bloków: Preambuły, Komendy, Danych oraz sumy kontrolnej CRC. Dla danych przesyłanych w kierunku od komputera do sterownika każdy blok jest wielkości jednego bajtu natomiast dla danych przesyłanych w kierunku od sterownika do komputera blok danych zajmuje cztery bajty, a pozostałe bloki po jednym bajcie.

Komputer-Sterownik

- KAT - jest to komenda informująca sterownik aby ustawił kąt obrotu serwomechanizmu na wartość z bloku DANE oraz odesłał komendę KAT-OK.

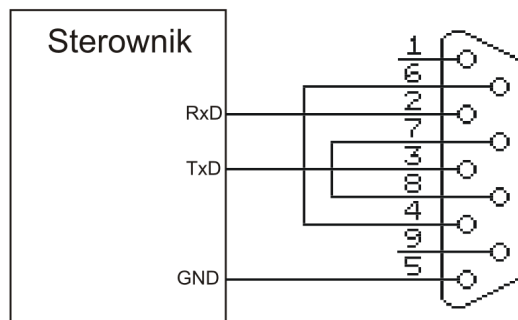


Rysunek 7: Schemat sterownika

- INICJACJA - jest to komenda informująca sterownik aby odesłał komendę INICJACJA-OK. Dzięki temu komputer wie czy sterownik jest podłączony do portu szeregowego, oraz moduł sonaru działa poprawnie.
- POMIAR - informacja dla sterownika aby wykonał pomiar oraz odesłał komendę POMIAR-OK
- POWTORZ - informuje sterownik że w poprzedniej ramce nie zgadza się suma kontrolna i należy wysłać ją ponownie.

Sterownik - Komputer

- KAT-OK - potwierdza ustawienie kąta na zadaną wartość
- INICJACJA-OK - informuje komputer że sterownik jest podłączony oraz transmisja jest możliwa i przekazuje wersję oprogramowania sterownika: Wers-num - oznacza wersję, natomiast Wers-pop - wartość wersji po kropce (np. Wers-num=1, Wers-pop=1, wersja 1.1).



Rysunek 8: Schemat połączenia gniazda DB9 ze sterownikiem

Ramka dla transmisji Komputer ==> Sterownik



Rysunek 9: Ramka dla kierunku komputer - sterownik

- **POMIAR-OK** - informuje komputer że pomiar został wykonany oraz przekazuje wyniki pomiaru: LewyM - starsza część 16-bitowego słowa reprezentującego odległość z lewego kanału sonaru, LewyL - młodsza część, PrawyM - starsza część 16-bitowego słowa reprezentującego odległość z prawego kanału sonaru, PrawyL - młodsza część
- **POWTORZ** - informuje komputer że w poprzedniej ramce nie zgadza się suma kontrolna i należy wysłać ją ponownie.

Preambuła Preambuła realizuje funkcję wyłącznie rozpoznania początku ramki, natomiast nie jest wykorzystana do synchronizacji transmisji. Przesyłane dane są na tyle krótkie że transmisja nie wymaga synchronizacji. Wartość tego bloku jest stała i wynosi $55h = 01010101b$

Ochrona przed błędami Pomimo że transmisja w obie strony odbywa się w trybie półdupleksowym, przy niewielkiej i stałej ilości danych, konieczne jest zastosowanie mechanizmów chroniących transmisję przed błędami.

Komputer Gdy komputer po wysłaniu ramki do sterownika nie otrzyma informacji zwrotnej do czasu timeout, uznaje to za błąd transmisji i ponawia wysyłanie. Po pięciu nieudanych próbach z rzędu zostaje wyświetlony komunikat o błędzie i transmisja zostaje przerwana. Jeśli komputer otrzyma błędną ramkę od sterownika (niezgodna suma CRC) wysyła komendę ponownie. W przypadku kiedy pięć kolejnych ramek zawiera błąd CRC, komputer przerywa transmisję i w oknie "info" wyświetla się komunikat o błędzie.

Nazwa	Komenda	Dane
KAT	0Ah	0 – 180
INICJACJA	0Bh	-
POMIAR	0Ch	-
POWTORZ	0Dh	-

Tablica 2: Spis komend i wartości danych dla transmisji komputer - sterownik

Ramka dla transmisji Sterownik => Komputer



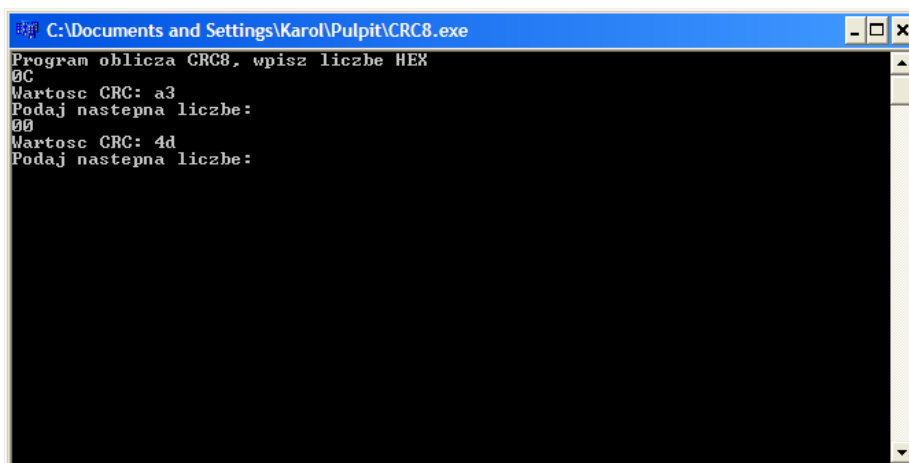
Rysunek 10: Ramka dla kierunku sterownik - komputer

Sterownik Sterownik po uruchomieniu czeka na nadejście danych z komputera. Po odebraniu bajtu, sprawdza czy jest on preambułą, jeśli tak, zaczyna liczyć kolejne bajty, jeśli nie - bajt jest ignorowany. W przypadku kiedy sterownik zliczy 4 bajty uznaje ramkę za kompletną i sprawdza sumę kontrolną. Jeśli ta się zgadza wykonuje polecenie, jeśli nie, wysyła komendę "Powtórz". Po odebraniu każdego bajtu po preambule oczekiwany jest pewien czas, jeżeli osiągnie on wartość timeout - ramka jest uznawana za błędną. Czas timeout został dobrany tak że jest równy czasowi transmisji kilkudziesięciu (60-70) bajtów. Gwarantuje to że tylko kompletne ramki składające się z czterech bajtów (dalsze bajty są ignorowane aż do kolejnej preambuły) i rozpoczynające się preambułą, zostaną poddane dalszej analizie.

Suma kontrolna CRC Ostatni blok w obu ramkach stanowi ośmio-bitowa cykliczna suma redundancyjna (CRC-8). Wielomian opisujący sumę: $X^8 + X^5 + X^4 + 1$. Jest to popularny typ sumy kontrolnej, stosowany m.in. w interfejsie 1-Wire™ firmy Dallas-Maxim. Do generowania wartości CRC posłużono się metodą tablicową. Dodatkowo napisano prosty program, umożliwiający obliczanie CRC na podstawie dostarczanych kolejnych bajtów.

NAZWA	KOMENDA	DANE1	DANE2	DANE3	DANE4
KAT-OK	1Ah	-	-	-	-
INICJACJA-OK	1Bh	Wers-num	Wers-pop	-	-
POMIAR-OK	1Ch	LewyM	LewyL	PrawyM	PrawyL
POWTORZ	1Dh	-	-	-	-

Tablica 3: Spis komend i wartości danych dla transmisji komputer - sterownik



Rysunek 11: Widok okna programu pomagającego obliczyć CRC

Tablica zastosowana do obliczeń:

TablicaCRC[256]= [0, 94, 188, 226, 97, 63, 221, 131, 194, 156, 126, 32, 163, 253, 31, 65, 157, 195, 33, 127, 252, 162, 64, 30, 95, 1, 227, 189, 62, 96, 130, 220, 35, 125, 159, 193, 66, 28, 254, 160, 225, 191, 93, 3, 128, 222, 60, 98, 190, 224, 2, 92, 223, 129, 99, 61, 124, 34, 192, 158, 29, 67, 161, 255, 70, 24, 250, 164, 39, 121, 155, 197, 132, 218, 56, 102, 229, 187, 89, 7, 219, 133, 103, 57, 186, 228, 6, 88, 25, 71, 165, 251, 120, 38, 196, 154, 101, 59, 217, 135, 4, 90, 184, 230, 167, 249, 27, 69, 198, 152, 122, 36, 248, 166, 68, 26, 153, 199, 37, 123, 58, 100, 134, 216, 91, 5, 231, 185, 140, 210, 48, 110, 237, 179, 81, 15, 78, 16, 242, 172, 47, 113, 147, 205, 17, 79, 173, 243, 112, 46, 204, 146, 211, 141, 111, 49, 178, 236, 14, 80, 175, 241, 19, 77, 206, 144, 114, 44, 109, 51, 209, 143, 12, 82, 176, 238, 50, 108, 142, 208, 83, 13, 239, 177, 240, 174, 76, 18, 145, 207, 45, 115, 202, 148, 118, 40, 171, 245, 23, 73, 8, 86, 180, 234, 105, 55, 213, 139, 87, 9, 235, 181, 54, 104, 138, 212, 149, 203, 41, 119, 244, 170, 72, 22, 233, 183, 85, 11, 136, 214, 52, 106, 43, 117, 151, 201, 74, 20, 246, 168, 116, 42, 200, 150, 21, 75, 169, 247, 182, 232, 10, 84, 215, 137, 107, 53];

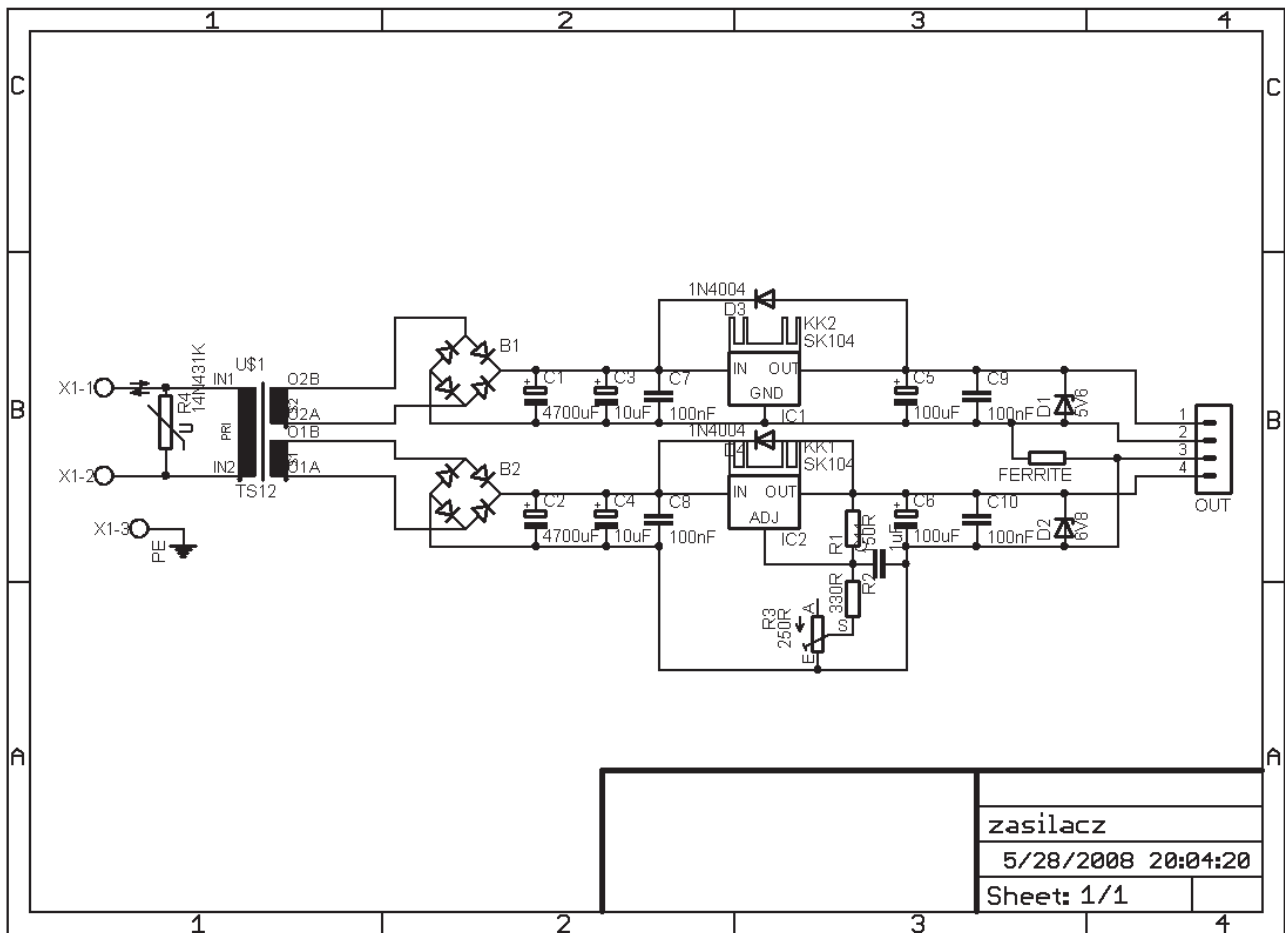
Funkcja wyliczająca: $CRC = TablicaCRC[CRC \text{ XOR bajt}]$;

2.4 Serwo

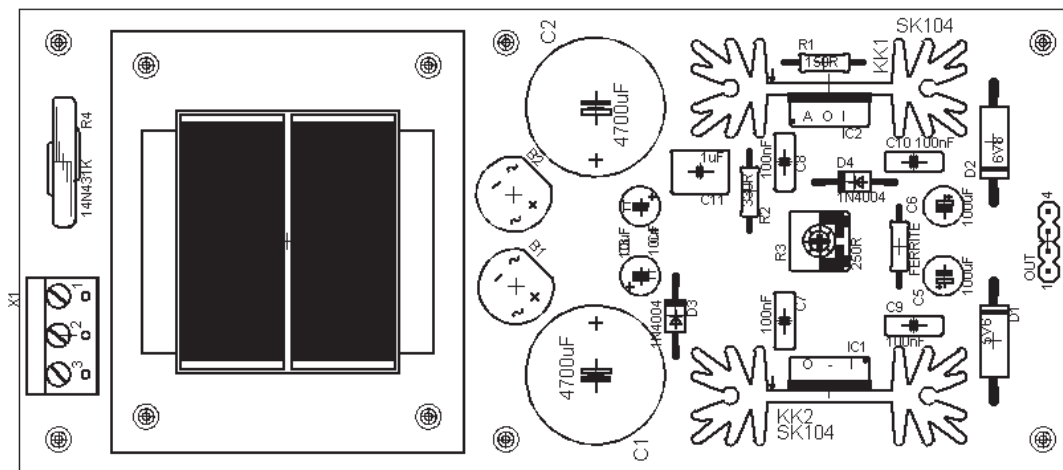
Serwomechanizm umożliwia obrót sonaru wokół osi pionowej w zakresie około 180° . Zastosowano serwomechanizm firmy Acorns. Rozdzielczość z jaką nastawiane jest serwo wynosi 1°

2.5 Zasilacz

Zasilacz dostarcza wysokostabilnego napięcia zasilania 5V dla sterownika oraz sonaru. Poza tym 4-6V dla zasilania serwomechanizmu. Ze względu na duże wymagania sonaru odnośnie jakości napięcia, zastosowano separację galwaniczną zasilania serwa i sonaru. Połączono masy w jednym punkcie za pomocą koralka ferrytowego. Napięcie dla serwa można regulować za pomocą potencjometru w zakresie 4-6V. W celu eliminacji zakłóceń elektromagnetycznych trafo zostało osłonięte za pomocą uziemionej metalowej siatki. Przy stabilizatorach umieszczono spore radiatory. Maksymalna wydajność napięcia zasilania jest szacowana na 1A dla serwa i 1A dla sterownika. Wejście zasilacza zabezpieczone jest bezpiecznikiem zwłocznym oraz warystorem. Wyjście zabezpieczone jest przez przekroczeniem dopuszczalnych napięć za pomocą 5-watowych diód zenera.



Rysunek 12: Schemat zasilacza



Rysunek 15: Rozmieszczenie elementów zasilacza

Lista elementów modułu sonaru

Element Wartość Obudowa Uwagi

C1 100n C0805
C2 100n C0805
C3 100n C0805
C4 10n C0805
C5 100n C0805
C6 100n C0805
C7 100n C0805
C8 10n C0805
C9 33p C0805
C10 33p C0805
C11 1u C0805
C12 1u C0805
C13 1u C0805
C14 1u C0805
C15 100n C0805
C16 100n C0805
C17 100n C0805
C18 1u C0805
C19 100n C0805
C20 100u SMD B
D1 Z5V6 MELF Dioda zenera 5v6
IC1 TL084D S014
IC2 TL084D S014
IC3 ATMEGA48-16AU TQFP32-08
IC4 MAX232ECWE S016
LED1 CHIPLD 1206 kolor niebieski
LED2 CHIPLD 1206 kolor niebieski
Q1 16MHz HC49/S niski z podkładką
Q2 BC807 SOT23 BEC
R1 2k2 R0805
R2 22k R0805
R3 1k R0805
R4 1k R0805
R5 22k R0805
R6 1K R0805
R7 1K R0805
R8 150k R0805
R9 1K2 R0805
R10 62K R0805
R11 2k2 R0805
R12 22k R0805
R13 1k R0805
R14 22k R0805
R15 1K R0805
R16 1K R0805
R17 150k R0805

R18 1K2 R0805
 R19 62K R0805
 R20 1k R0805
 R21 10K R0805
 R22 500R R0805
 R23 500R R0805
 R24 1K2 R0805
 SV4 MA06-1 goldpin
 U\$1 40R F/CM12P odbiornik 12mm 40KHz
 U\$2 40R F/CM12P odbiornik 12mm 40KHz
 U\$3 40T F/CM12P nadajnik 12mm 40KHz

Lista elementów sterownika

Element	Wartość	Obudowa	Uwagi
C1	1u	C-EUC0805	
C2	1u	C-EUC0805	
C3	1u	C-EUC0805	
C4	1u	C-EUC0805	
C5	33n	C-EUC0805	
C6	100n	C-EUC0805	
C7	100n	C-EUC0805	
C8	100n	C-EUC0805	
C9	1u	C-EUC0805	
C11	10u	CPOL-EUTT2D4	
C12	100u	CPOL-EUTT2D5	
IC1	MC9S12A64	Moduł z mikroprocesorem	
IC2	MAX232ECWE	S016	
JP1	PINHD-1X2	złącze goldpin 1x2	
JP2	PINHD-1X3	złącze goldpin 1x3	
POWER	MA04-1	złącze 4-pin	
R1	56R	R-EU_M0805	
R2	1K	R-EU_M0805	
R3	1K	R-EU_M0805	
R4	1K	R-EU_M0805	
RS232	MA05-2	złącze goldpin 2x5	
SERVO	MA03-1	złącze goldpin 1x3	
SONAR	MA07-1	złącze 7-pin	

Lista elementów zasilacza

Element	Wartość	Obudowa	Uwagi
B1		RB1A	mostek prostowniczy 1.5A 50V
B2		RB1A	
C1	4700uF	E7,5-18	
C2	4700uF	E7,5-18	

C3	10uF	TT2D5
C4	10uF	TT2D5
C5	100uF	E5-5
C6	100uF	E5-5
C7	100nF	C050-025X075
C8	100nF	C050-025X075
C9	100nF	C050-025X075
C10	100nF	C050-025X075
C11	1uF	C050-055X075
D1	5V6	C1702-15 dioda zenera 5W
D2	6V8	C1702-15
D3	1N4004	D041-10
D4	1N4004	D041-10
FERRITE	0207/10	Koralik ferrytowy
IC1	LM2940-5V	T0-220
IC2	LM1086-Adj	T0-220
KK1	SK104	radiator
KK2	SK104	
OUT	MA04-1	złącze 4-pin
R1	150R	0207/10
R2	330R	0207/10
R3	250R	CA6V potencjometr
R4	14N431K	S14K11 warystor
U\$1	TS12	trafo TS12/008 2x7.5V
X1		Zacisk montażowy 3-pin